Hướng dẫn cài đặt môi trường

Mô tả sơ bộ về môi trường phát triển

JDK

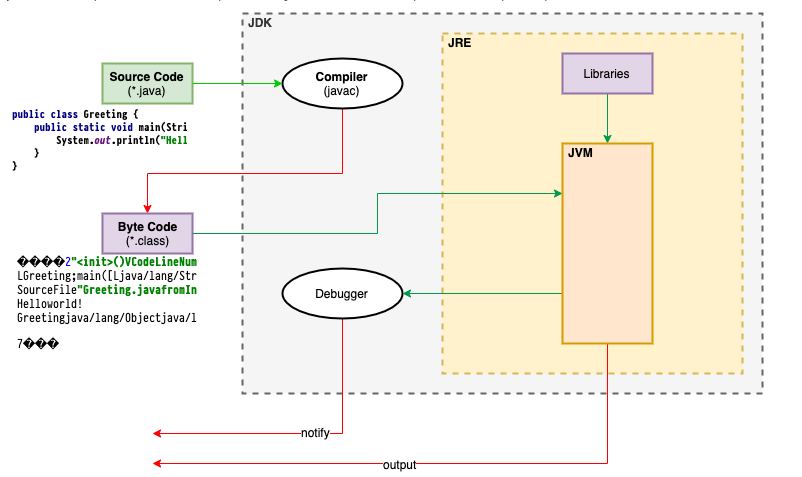
Trong chương này bạn học các kỹ thuật lập trình nâng cao, để có nền tảng lập trình vững vàng khi đi làm. Cũng như chuẩn bị các kiến thức và kỹ năng cần thiết để học về lập trình phần-mềm-web-backend.

Phần mềm mà một nhà phát triển *phần mềm Java* tạo nên là *một tập hợp các tập tin byte-code mà Máy ảo Java (JVM) có thể đọc và thực thi được*. Bất kỳ đối tượng nào muốn chạy phần mềm Java đều cần tới JVM và các phần mềm hỗ trợ của nó. Bộ phần mềm này được đóng gói dưới tên *môi-trường-thực-thi-Java (JRE)*.

Nhà phát triển sẽ không trực tiếp viết phần mềm lên file byte-code, bởi nó không dễ đọc với con người. Thay vào đó, họ viết mã dưới cú pháp của ngôn ngữ lập trình Java. Những mã này được dịch thành mã byte-code bởi phần mềm java compiler.

Người dùng bình thường không cần tới java compiler, bởi thứ họ nhận được đã là phần mềm chạy được rồi. Bộ các công cụ bổ sung dành riêng cho nhà phát triển được đóng gói kèm với JRE để thành *bộ-công-cụ-phát-triển-Java* (JDK).

JDK là điều kiện tối thiểu để nhà phát triển Java có thể bắt đầu phát triển được sản phẩm.



IDE

Rất nhiều khâu trong giai đoạn lập trình là lặp đi lặp lại hoặc rất rắc rối. Ví dụ như chạy trình dịch, dò lỗi, kiểm tra chính tả, chạy thử chương trình, commit mã… Để nhà phát triển bỏ bớt các thao tác thừa và có năng lượng tập trung vào giải quyết vấn đề thực sự, họ cần tới một môi trường hỗ trợ phát triển đủ tối tân.

Các IDE là những phần mềm có khả năng tích hợp tính năng của nhiều phần mềm khác cùng tham gia vào việc phát triển lại với nhau, dưới một thiết kế dễ dàng sử dụng. Qua đó tăng cao hiệu suất làm việc của nhà phát triển.

Để bạn có thể tập trung vào khía cạnh chính trong module - là các kỹ năng lập trình cấp cao, một IDE tốt là cần thiết.

Hướng dẫn cài đặt

JDK

**Chọn phiên bản**

Trước tiên, có nhiều phiên bản JRE, kèm theo đó là các phiên bản JDK tương ứng. Việc sử dụng phiên bản JDK quá mới không phải là điều tốt. Các công cụ trong đó có khả năng sẽ không thể chạy được các phần mềm cũ. Với người học, sử dụng phiên bản mới sẽ tăng tỷ lệ các tài liệu nhiễu hiện lên trong lúc học. Phiên bản phù hợp để học trong module này là JDK 1.7.

Cùng một phiên bản, JRE và JDK có thể do nhiều tổ chức viết ra những phiên bản khác nhau, ví dụ phiên bản của Oracle, hay của các nhóm opensource khác. Việc sử dụng các phiên bản khác nhau về cơ bản không gây ra sự thay đổi về hành vi của phần mềm, nhưng có thể khác nhau về hiệu năng hay số lỗi.

**Chọn phương án cài đặt**

Về cơ bản, việc cài đặt JDK nghĩa là làm cho tất cả các phần mềm trong bộ công cụ này nằm ở trạng thái sử dụng được. Nghĩa là:

1. Trong hệ thống có các phần mềm cùng các tập tin hỗ trợ của chúng
2. Tất cả các phần mềm đều được khai báo vào biến môi trường $PATH của hệ thống, nhờ đó chúng có thể được gọi tại bất cứ shell nào.

Để đơn giản hóa quá trình cài đặt, người ta thường sử dụng các phần mềm có khả năng làm được hai việc trên, ví dụ như các chương trình *\*.msi*, *apt-get*, *yum*… Với nhà phát triển Java, việc này không được khuyến khích, bởi:

* Nhà phát triển thật ra đã ủy quyền việc tìm đến và triệu gọi JRE cho IDE, nên việc (2) là không cần thiết
* Nhà phát triển thật ra thường cần sử dụng nhiều hơn một JDK, để phát triển dự án khác, nghiên cứu, học… Nên việc cài đặt một phiên bản cố định là không thích hợp.

**Sử dụng Java version management**

Có những chương trình có khả năng điều phối môi trường Java trong hệ thống, cho phép tải, cài đặt, và quản trị nhiều phiên bản JRE/JDK cùng một lúc. Chúng được gọi là *Java version management*. Một trong số các jvm (phân biệt với JVM) phổ biến là [*jabba*](https://github.com/shyiko/jabba).

Để cài đặt jabba trên macOS/Linux, sử dụng command line sau:

curl -sL https://github.com/shyiko/jabba/raw/master/install.sh | bash && . ~/.jabba/jabba.sh

Để cài đặt trên Windowns 10, bật trình *Powershell* và sử dụng command line sau:

[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12  
Invoke-Expression (  
 Invoke-WebRequest https://github.com/shyiko/jabba/raw/master/install.ps1 -UseBasicParsing  
).Content

Lưu ý: hướng dẫn cài đặt có thể thay đổi theo thời gian, hãy tham khảo tài liệu tại trang chủ.

Sau đó có thể sử dụng jabba để cài đặt và ấn định môi trường Java thích hợp:

# liệt kê các phiên bản Java mà jabba có thể thao túng  
jabba ls-remote  
   
# cài đặt Oracle JDK  
jabba install 1.8  
# cài đặt Zulu OpenJDK  
jabba install zulu@1.8  
# cài đặt OpenJDK  
jabba install openjdk@1.11  
   
# gỡ bỏ JDK  
jabba uninstall zulu@1.6.77  
   
# kết nối tới JDK có sẵn trên hệ thống  
jabba link system@1.8.72 /Library/Java/JavaVirtualMachines/jdk1.8.0\_72.jdk  
   
# liệt kê tất cả các bản JDK đang được thao túng  
jabba ls  
   
# chuyển đổi môi trường Java của hệ thống sang phiên bản chỉ định  
jabba use adopt@1.8  
jabba use zulu@~1.6.97  
   
# có thể đặt mặc định một phiên bản Java làm môi trường hệ thống  
jabba alias default 1.8

**Lưu ý**: mặc định, các tập tin của JDK được jabba tải về và đặt tại thư mục ~/.jabba/jdk, hãy lưu ý điều này để khai báo cho IDE.

jabba  
├── bin  
│ └── jabba  
├── default.alias  
├── jabba.fish  
├── jabba.sh  
└── jdk  
 ├── 1.6.65  
 ├── default -> /Users/\_/.local/share/jabba/jdk/zulu@1.11.0  
 ├── zulu@1.11.0  
 ├── zulu@1.7.201  
 ├── zulu@1.8.212  
 └── zulu@1.9.0

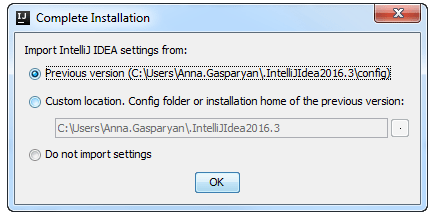
IDE

Tại đây hướng dẫn cài đặt [JetBrains ***IntelliJ*** IDEA](https://www.jetbrains.com/idea/) - một trong số các IDE đầu bảng trong giới nhà phát triển Java. IDEA có sự tối tân, hiện đại, ổn định, và có trải nghiệm sử dụng đồng bộ xuyên suốt.

Bạn có thể cài đặt phần mềm bằng cách download tại trang chủ của hãng. Nếu sử dụng nhiều hơn một công cụ của JetBrains, nên cân nhắc cài đặt thông qua [JetBrains Toolbox](https://www.jetbrains.com/toolbox-app/). Quá trình cài đặt rất đơn giản, sau khi cài xong, hãy mở IDEA lên lần đầu tiên.

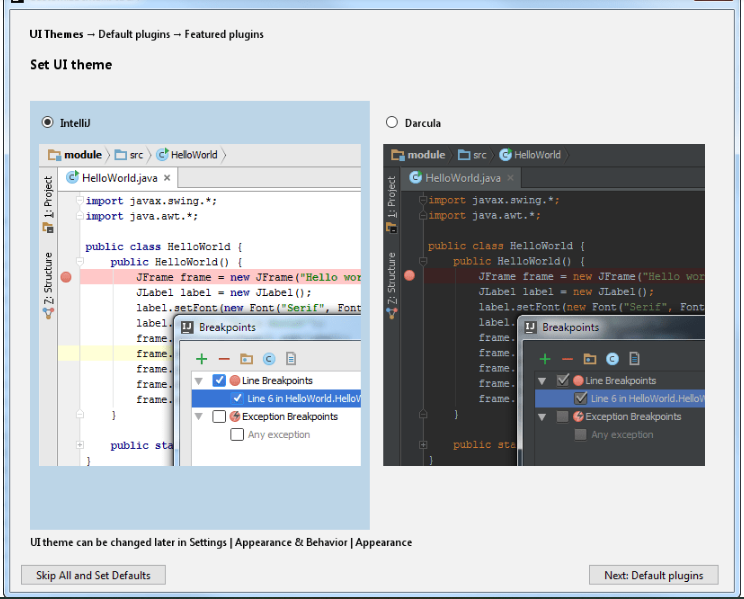
**Dùng lại cấu hình cũ nếu có**

Nếu bạn đang có sẵn một phiên bản IDEA cũ hơn trên hệ thống, và muốn dùng lại các cấu hình có sẵn, hãy sử dụng các tùy chọn liên quan đến dùng lại cấu hình. Nếu không, chọn *Do not import settings*.



**Lựa chọn tông màu**

Nếu màn hình máy tính của bạn đủ tốt trên 3 yếu tố: màu sắc, độ tương phản, độ sáng, bạn có thể sử dụng tông màu tối. Nếu không, bạn nên chọn tông màu sáng.



**Gỡ bỏ hoặc cài đặt thêm các plugins**

Tất cả các plugins đều có thể được cài thêm hoặc bỏ đi sau này. Hãy mạnh dạn bỏ đi những thứ không cần đến trong tương lai gần. Cấu hình khuyến khích như sau:

**Java Frameworks**: để lại Spring, Hibernate và JavaEE

**Build Tools**: chỉ để lại Gradle

**Web Development**: bật hết

**Version Controls**: chỉ để lại Git

**Test Tools**: chỉ để lại JUnits

**Application Servers**: chỉ để lại Tomcat

**Clouds**: tắt hết

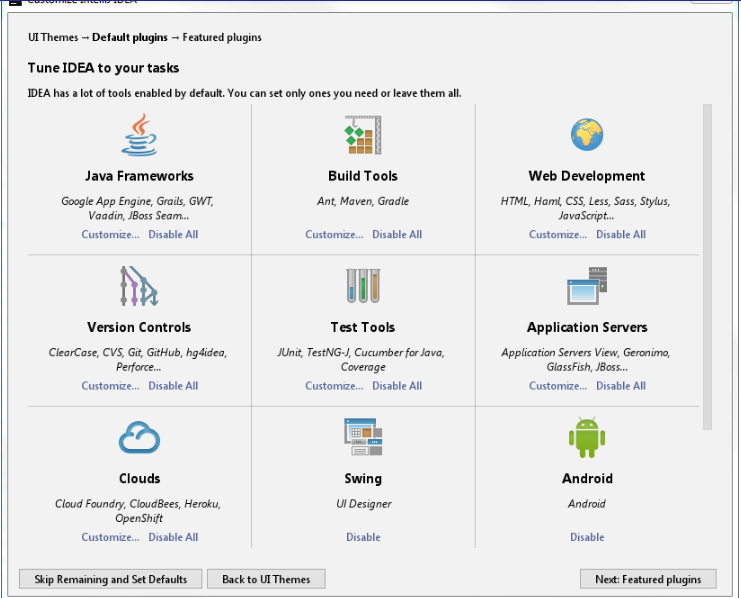
**Swing**: tắt

**Android**: tắt

**Database Tool**: để lại

**Other Tools**: để lại

**Plugins Development**: tắt



**Feature Flugins**

Feature Plugins là các tính năng không có sẵn một cách mặc định, tuy vậy bạn vẫn có thể tải về bất kỳ lúc nào. Chỉ dùng khi bạn biết mình đang muốn làm gì.

**Material Theme UI**: thay giao diện truyền thống thành giao diện theo ngôn ngữ thiết kế Material Design.

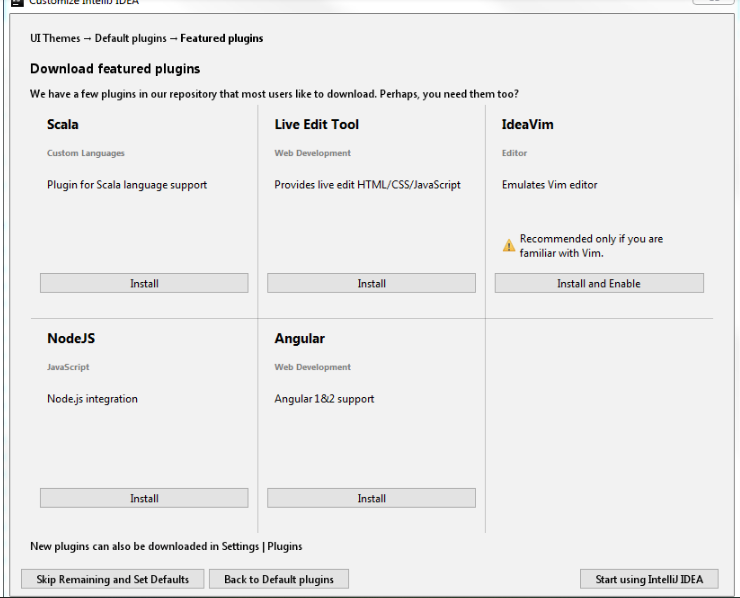
**Scalar**: dùng khi bạn lập trình Scalar

**Live Edit Tool**: dùng khi bạn thiết kế UI/UX mà muốn xem kết quả nhanh, không cần tới trình duyệt.

**IdeaVim**: giả lập vim editor.

**NodeJS**: dùng khi bạn muốn lập trình NodeJS

**Angular**: dùng khi bạn muốn lập trình Angular



Sau khi cấu hình, IDEA đã sẵn sàng sử dụng.